Homework #2

1. **Flowers**

Data: `flowers` dataset in **cluster** package

a. Rename the column names and recode the levels of categorical variables to descriptive names. For example, “V1” should be renamed “winters” and the levels to “no” or “yes”. Display the full dataset.

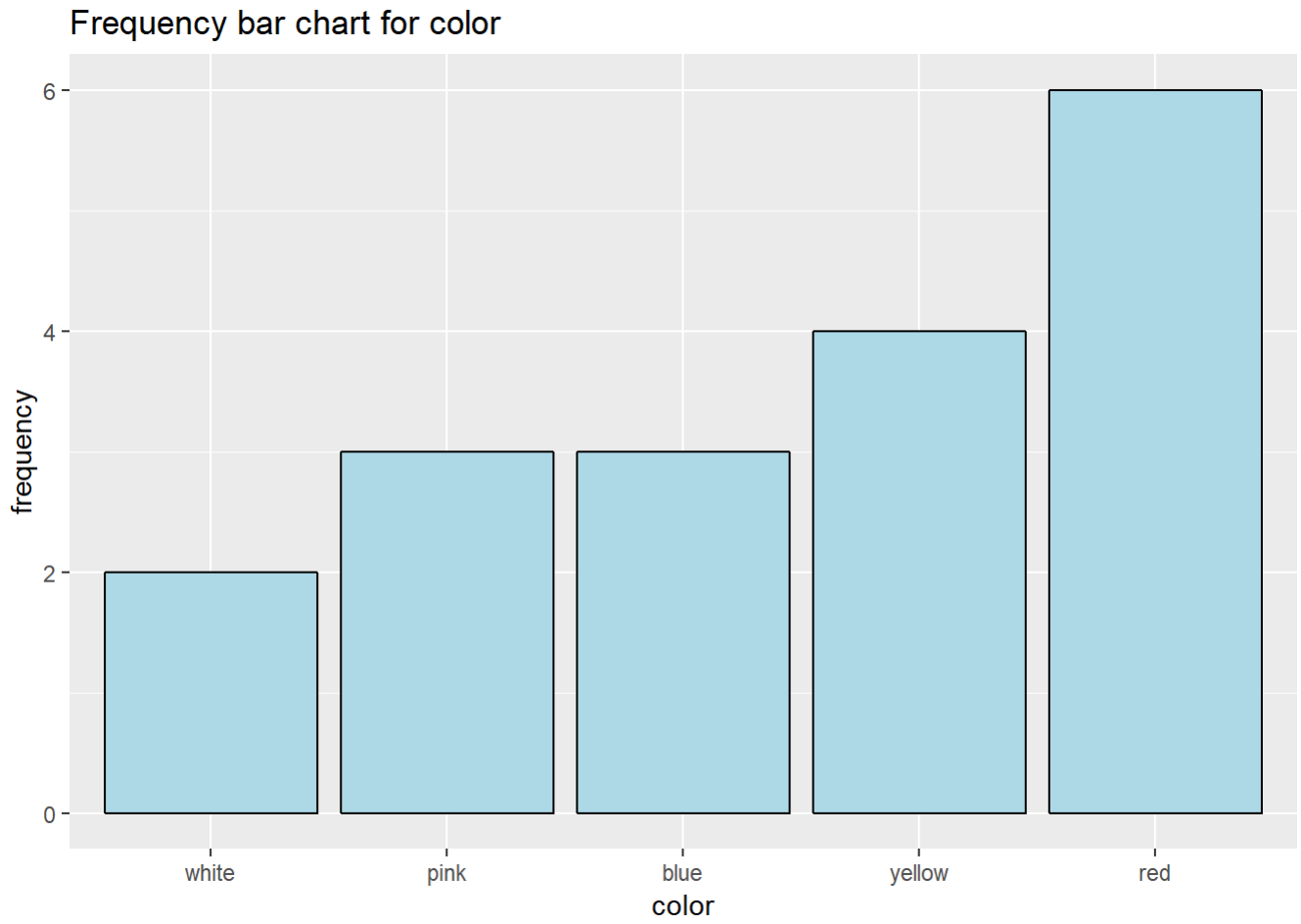
```
# copy to local variable as to not alter original dataset
my_flower <- flower
my_flower <- rename(my_flower, winters = V1, shadow = V2, tubers = V3, color = V4, soil = V5, pr
eference = V6, height = V7, distance = V8)
levels(my_flower$winters) <- factor(recode(levels(my_flower$winters),`0` = "no", `1` = "yes"))
levels(my_flower$shadow) <- factor(recode(levels(my_flower$shadow),`0` = "no", `1` = "yes"))
levels(my_flower$tubers) <- factor(recode(levels(my_flower$tubers),`0` = "no", `1` = "yes"))
levels(my_flower$color) <- factor(recode(levels(my_flower$color),`1` = "white", `2` = "yellow",
 `3` = "pink", `4` = "red", `5` = "blue"))
levels(my_flower$soil) <- factor(recode(levels(my_flower$soil),`1` = "dry",`2` = "normal", `3` =
 "wet"))
my_flower
```

winters <fctr>	shadow <fctr>	tubers <fctr>	color <fctr>	soil <ord>	preference <ord>	height <dbl>	distance <dbl>
no	yes	yes	red	wet	15	25	15
yes	no	no	yellow	dry	3	150	50
no	yes	no	pink	wet	1	150	50
no	no	yes	red	normal	16	125	50
no	yes	no	blue	normal	2	20	15
no	yes	no	red	wet	12	50	40
no	no	no	red	wet	13	40	20
no	no	yes	yellow	normal	7	100	15
yes	yes	no	pink	dry	4	25	15
yes	yes	no	blue	normal	14	100	60

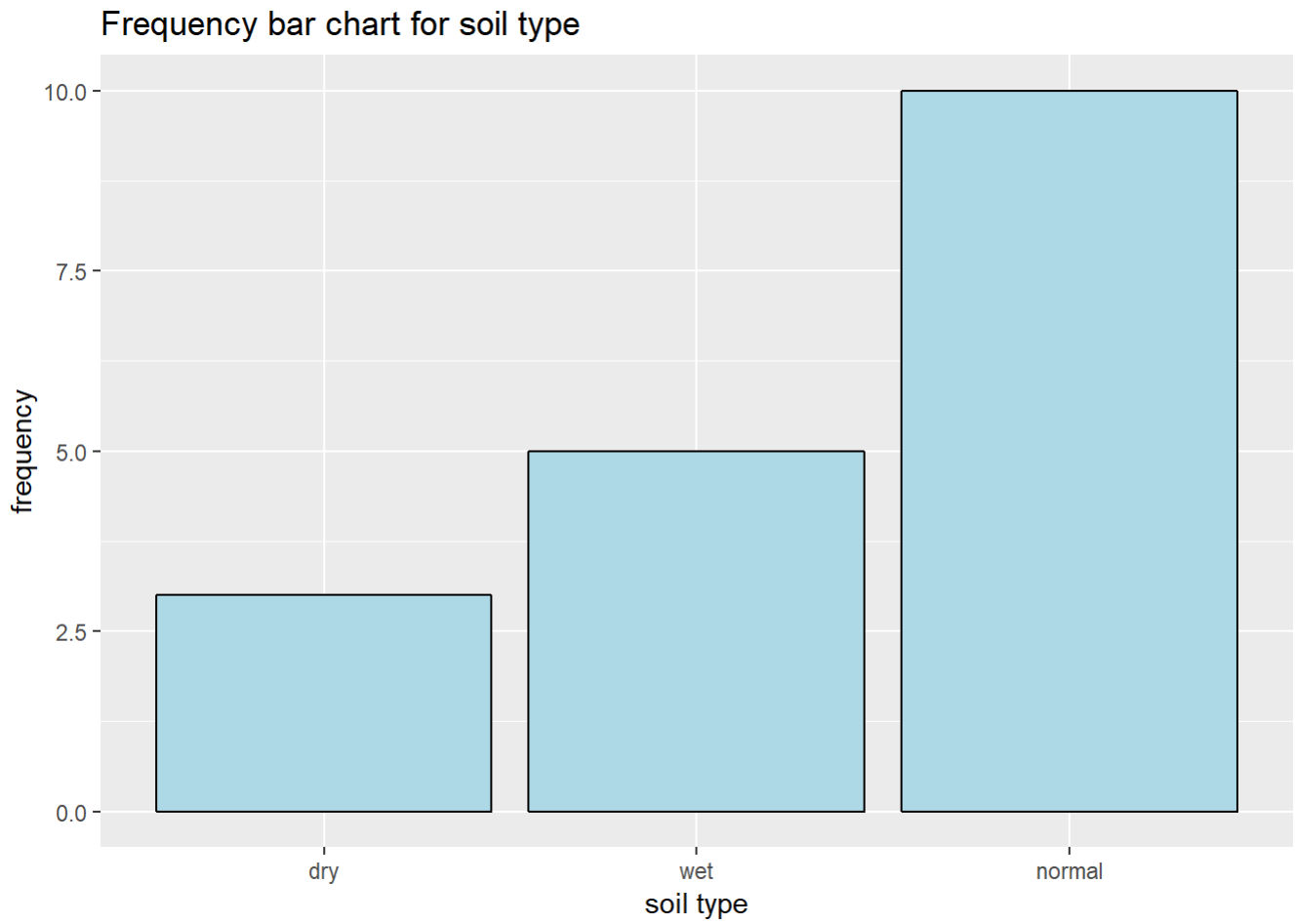
1-10 of 18 rows Previous **1** 2 Next

b. Create frequency bar charts for the `color` and `soil` variables, using best practices for the order of the bars.

```
color_count <- my_flower %>% group_by(color) %>% summarise(col_count = n())
soil_count <- my_flower %>% group_by(soil) %>% summarise(so_count = n())
ggplot(color_count, aes(reorder(color, col_count),col_count)) + geom_col(color = "black",fill =
"lightBlue")+ xlab("color")+ ylab("frequency")+ggtitle("Frequency bar chart for color")
```




```
# ordering based on frequency
ggplot(soil_count, aes(reorder(soil, so_count),so_count)) + geom_col(color = "black",fill = "lig
htBlue")+ xlab("soil type")+ ylab("frequency") +ggtitle("Frequency bar chart for soil type")
```

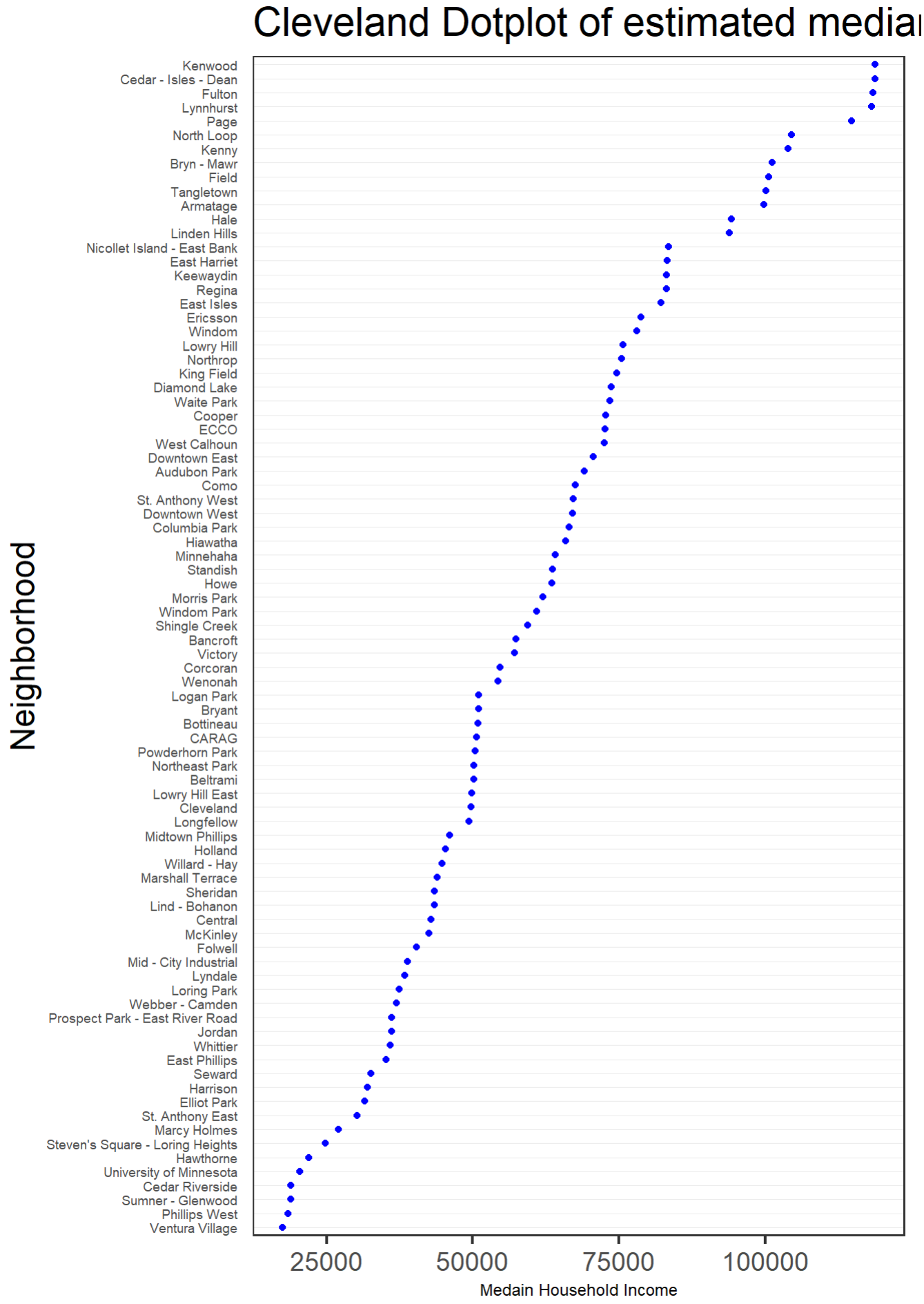


2. Minneapolis

Data: `MplsDemo` dataset in **carData** package

a. Create a Cleveland dot plot showing estimated median household income by neighborhood.

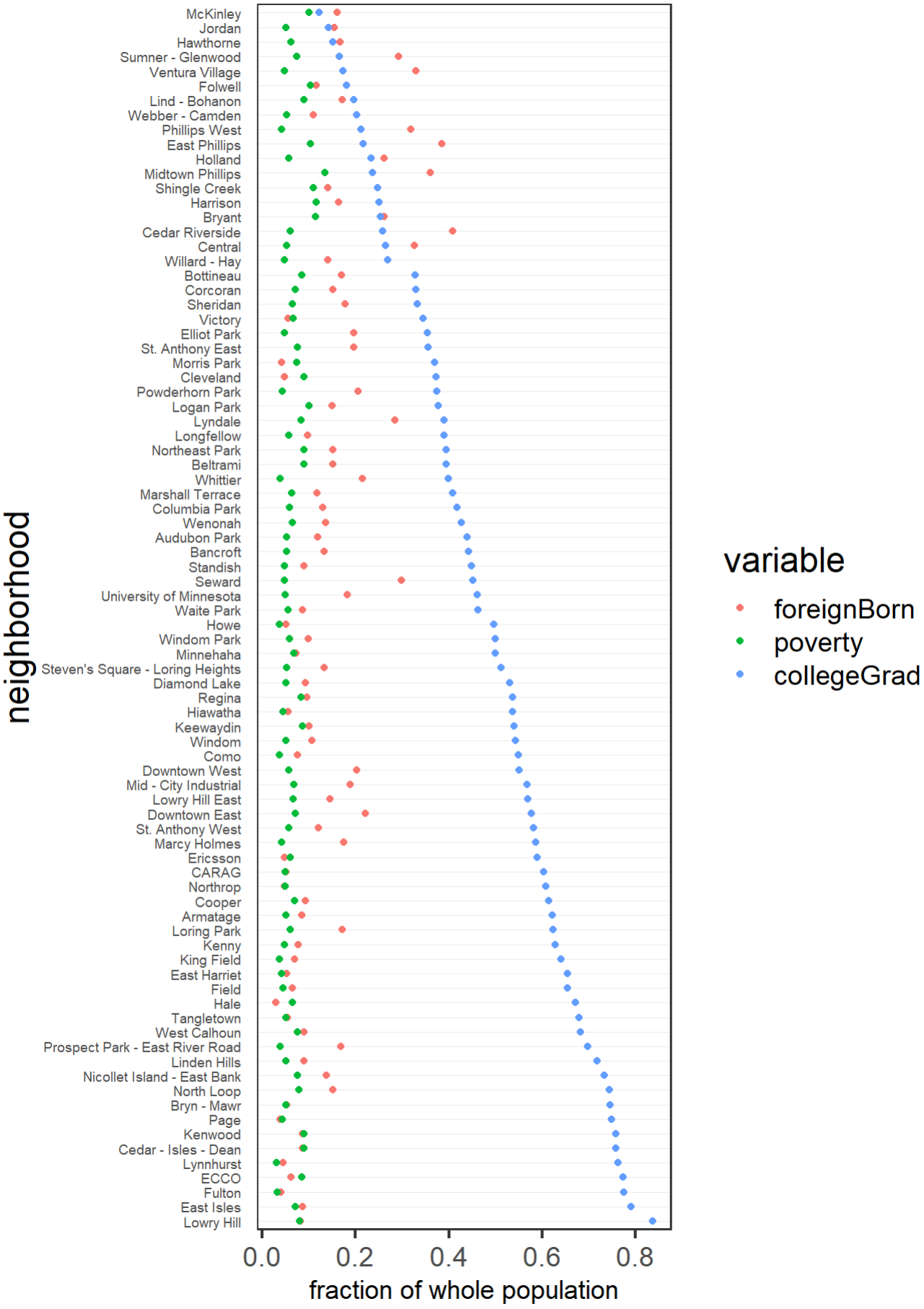
```
# theme_dotplot referred from lecture slides
theme_dotplot <- theme_bw(18) +
    theme(axis.text.y = element_text(size = rel(.50)),
          axis.ticks.y = element_blank(),
          axis.title.x = element_text(size = rel(.50)),
          panel.grid.major.x = element_blank(),
          panel.grid.major.y = element_line(size = 0.25),
          panel.grid.minor.x = element_blank())
ggplot(MplsDemo, aes(x = hhIncome, y = reorder(neighborhood,hhIncome)))+geom_point(color = "blu
e")+theme_dotplot+ggtitle("Cleveland Dotplot of estimated median household by neighborhood")+xla
b("Medain Household Income")+ylab("Neighborhood")
```


Cleveland Dotplot of estimated median
Neighborhood
Kenwood
Cedar - Isles - Dean
Fulton
Lynnhurst
Page
North Loop
Kenny
Bryn - Mawr
Field
Tangletown
Armatage
Hale
Linden Hills
Nicollet Island - East Bank
East Harriet
Keewaydin
Regina
East Isles
Ericsson
Windom
Lowry Hill
Northrop
King Field
Diamond Lake
Waite Park
Cooper
ECCO
West Calhoun
Downtown East
Audubon Park
Como
St. Anthony West
Downtown West
Columbia Park
Hiawatha
Minnehaha
Standish
Howe
Morris Park
Windom Park
Shingle Creek
Bancroft
Victory
Corcoran
Wenonah
Logan Park
Bryant
Bottineau
CARAG
Powderhorn Park
Northeast Park
Beltrami
Lowry Hill East
Cleveland
Longfellow
Midtown Phillips
Holland
Willard - Hay
Marshall Terrace
Sheridan
Lind - Bohanon
Central
McKinley
Folwell
Mid - City Industrial
Lyndale
Loring Park
Webber - Camden
Prospect Park - East River Road
Jordan
Whittier
East Phillips
Seward
Harrison
Elliot Park
St. Anthony East
Marcy Holmes
Steven's Square - Loring Heights
Hawthorne
University of Minnesota
Cedar Riverside
Sumner - Glenwood
Phillips West
Ventura Village
25000
50000
75000
100000
Medain Household Income

b. Create a Cleveland dot plot *with multiple dots* to show percentage of 1) foreign born, 2) earning less than twice the poverty level, and 3) with a college degree *by neighborhood.* Each of these three continuous variables should appear in a different color. Data should be sorted by college degree.

```
my_demo <- MplsDemo

# select the required columns and melt the data to get all the 3 variables on the same plot
my_demo <- MplsDemo %>% select(neighborhood,foreignBorn,poverty,collegeGrad)
melt_demo <- melt(my_demo)

# plot by sorting based on college grad
ggplot(melt_demo,aes(x=value,y=fct_reorder2(neighborhood,sort(variable),value),color=variable))+
geom_point()+theme_dotplot+xlab("fraction of whole population")+ylab("neighborhood") + ggtitle(
"Multiple variables on one plot")
```

Multiple variables on one plot



c. What patterns do you observe? What neighborhoods do not appear to follow these patterns?

Observed Pattern: Percent of poverty < Percent of foreign born < percent of College Grad

There are many neighborhoods that do not follow the pattern. To list a few

Jordan Hawthorne McKinley Victory Morris Park Cleveland Cedar Riverside East Philips

3. Taxis

Data: NYC yellow cab rides in June 2018, available here:

http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml
(http://www.nyc.gov/html/tlc/html/about/trip_record_data.shtml)

It's a large file so work with a reasonably-sized random subset of the data.

Draw four scatterplots of `tip_amount` vs. `far_amount` with the following variations:

a. Points with alpha blending

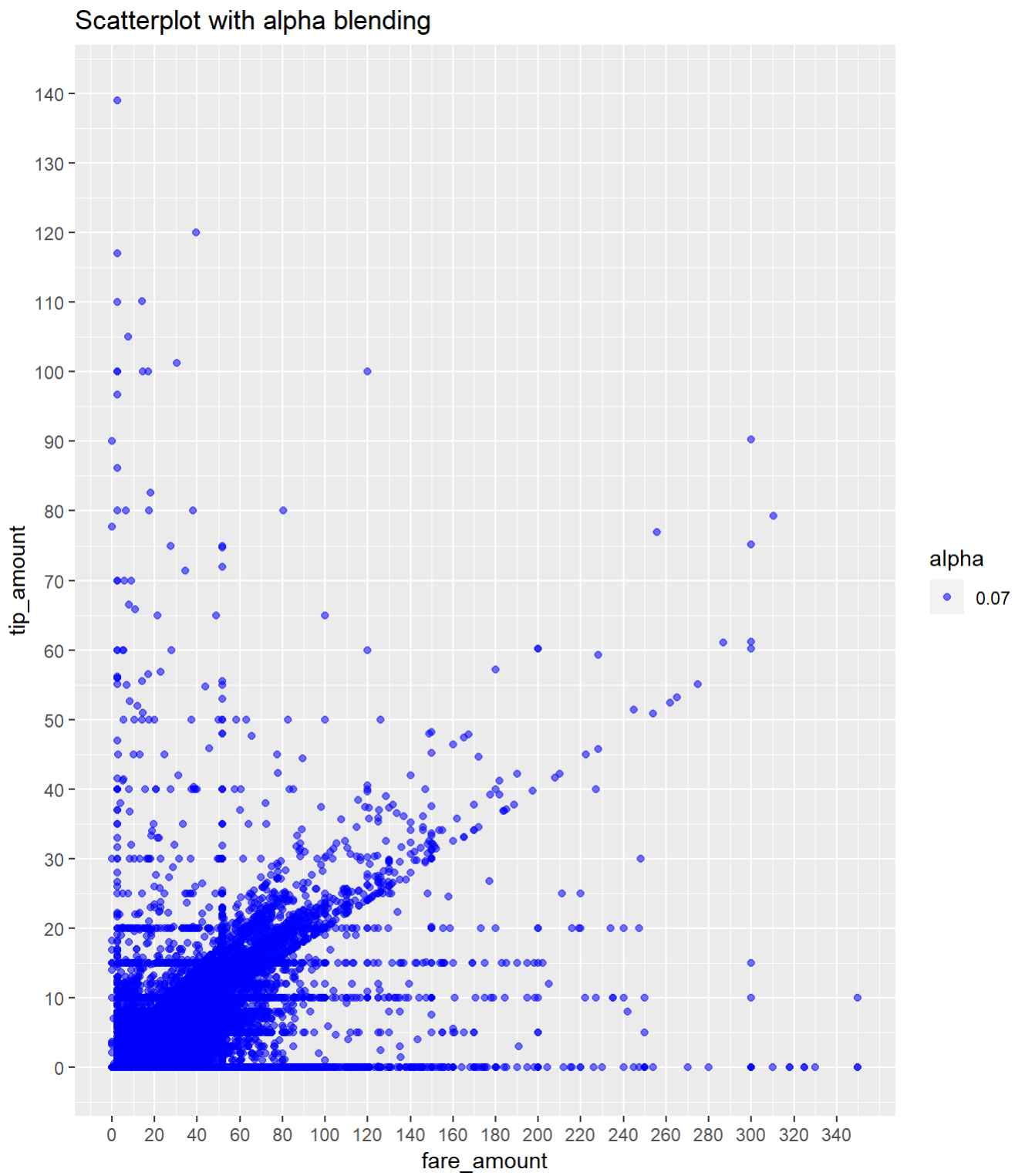
b. Points with alpha blending + density estimate contour lines

c. Hexagonal heatmap of bin counts

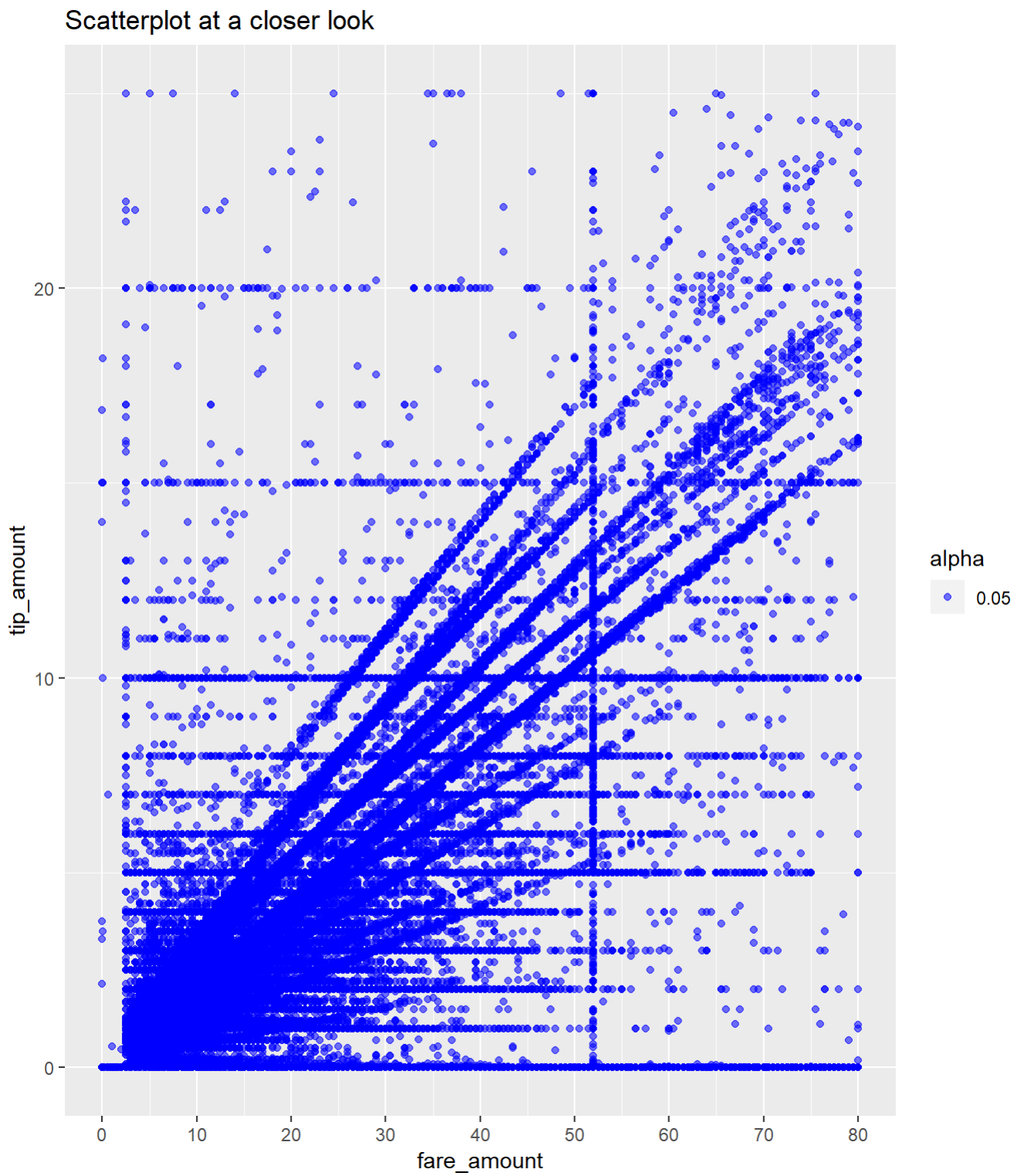
d. Square heatmap of bin counts

For all, adjust parameters to the levels that provide the best views of the data.

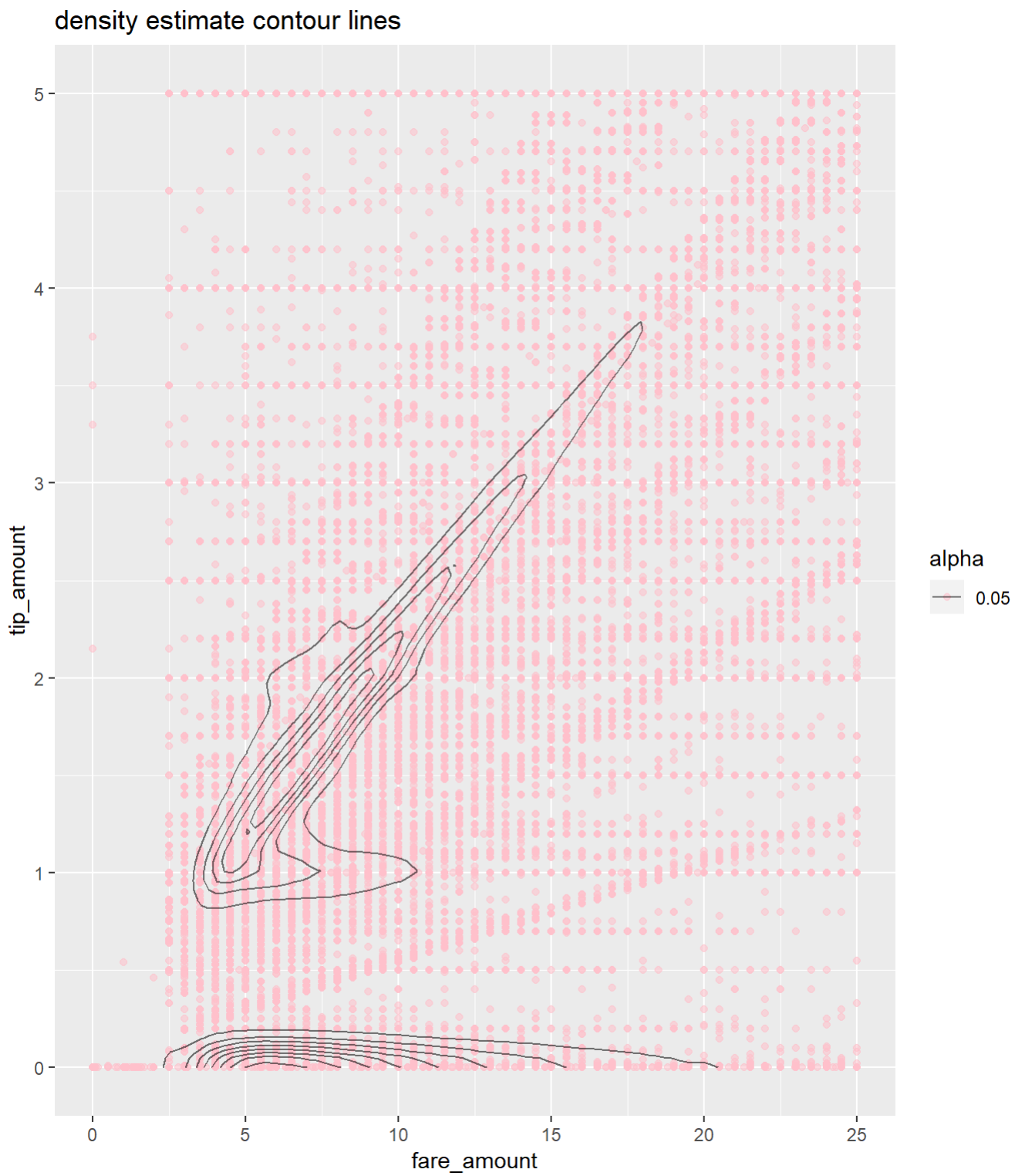
```
cab_data1 <- read.csv("C:/Users/aishw/Documents/cab_data.csv",colClasses = c(rep(x="NULL",10),"n
umeric",rep("NULL",2),"numeric",rep("NULL",3)))

rand_data1 <- sample_n(cab_data1,800000) %>% filter(tip_amount>=0 & fare_amount>=0)

ggplot(rand_data1, aes(fare_amount,tip_amount, alpha = 0.07)) +
geom_point(color = "blue")+scale_y_continuous(limits = c(0,140),breaks =
seq(0,140,10)) + scale_x_continuous(limits = c(0,350),breaks =
seq(0,350,20))+ggtitle("Scatterplot with alpha blending")
```



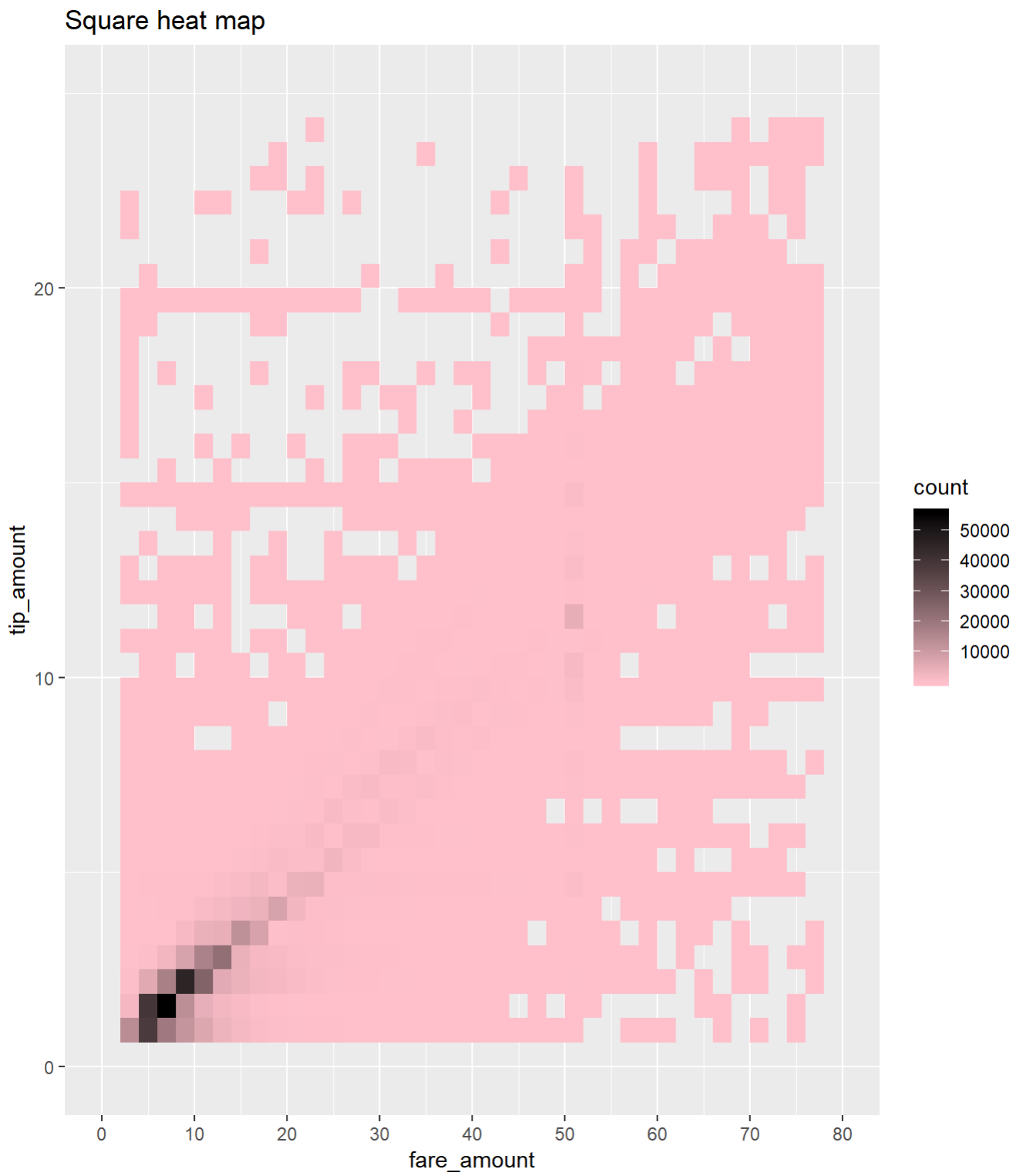

```
ggplot(rand_data1, aes(fare_amount,tip_amount, alpha = 0.05)) +
geom_point(color = "blue")+scale_y_continuous(limits = c(0,25),breaks =
seq(0,25,10)) + scale_x_continuous(limits = c(0,80),breaks =
seq(0,80,10))+ggtitle("Scatterplot at a closer look")
```



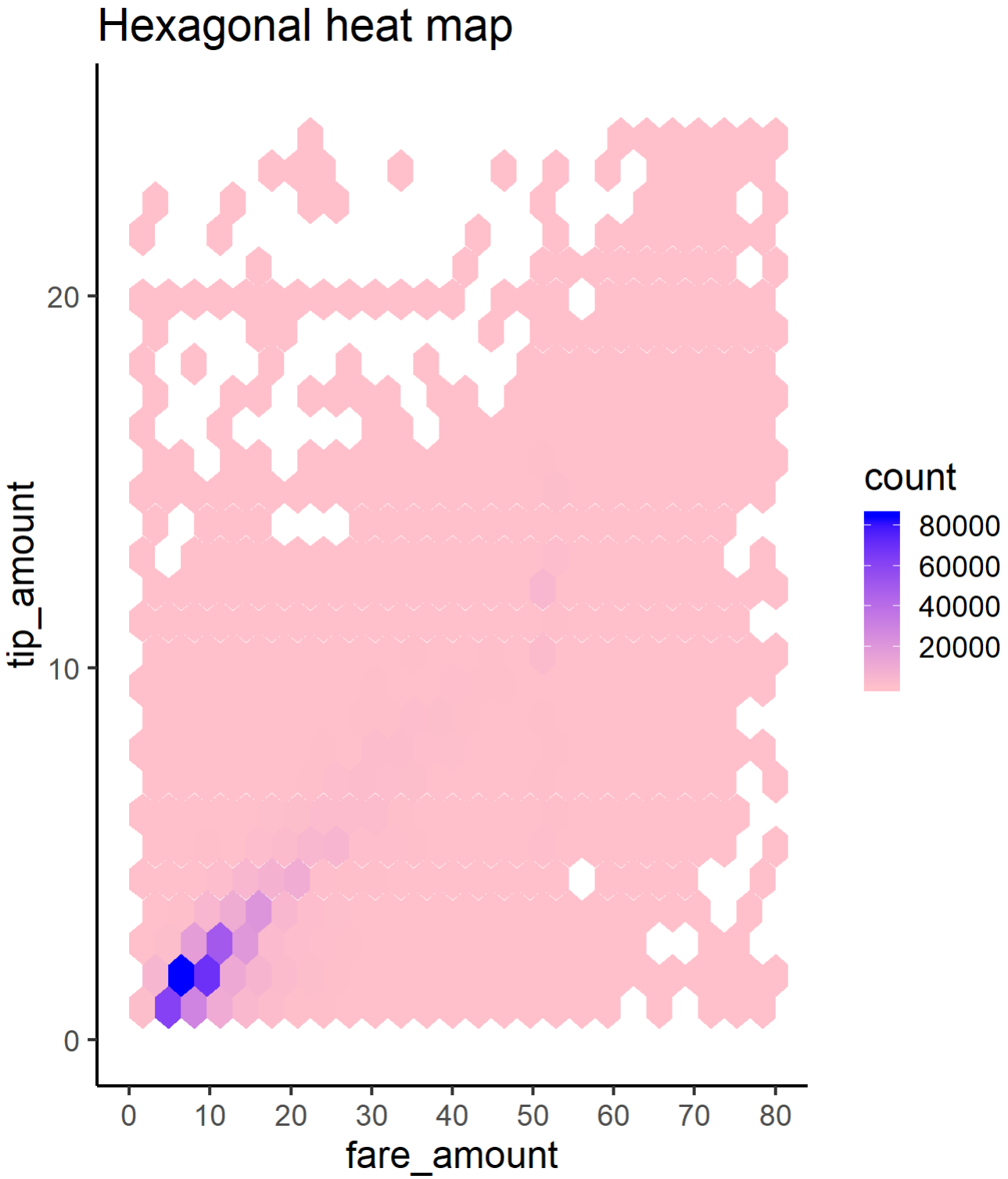

```
ggplot(rand_data1, aes(fare_amount,tip_amount, alpha = 0.05))+
geom_point(color = "pink")  +  xlim(0,25) + ylim(0,5) +
geom_density_2d(color="black")+ggtitle("density estimate contour lines")
```




```
ggplot(rand_data1, aes(fare_amount,tip_amount)) + scale_fill_gradient(low = "pink", high = "blac
k")+
geom_bin2d(bins = 40)+scale_y_continuous(limits = c(0,25),breaks =
seq(0,25,10)) + scale_x_continuous(limits = c(0,80),breaks =
seq(0,80,10)) + ggtitle("Square heat map")
```




```
ggplot(rand_data1, aes(fare_amount,tip_amount)) + geom_hex(bins = 25) +
scale_fill_gradient(low = "pink", high = "blue") +  theme_classic(18) +
scale_y_continuous(limits = c(0,25),breaks =
seq(0,25,10)) + scale_x_continuous(limits = c(0,80),breaks =
seq(0,80,10))+ggtitle("Hexagonal heat map")
```



e. Describe noteworthy features of the data, using the “Movie ratings” example on page 82 (last page of Section 5.3) as a guide.

1. In many instances, people have not tipped.
2. In certain instances, the tip amount is very high (>$100) whereas the fare amounted to $0. These data points may be outliers.
3. There are very less number of instances where people with fare amounts >$220 have tipped
4. The tip amount increases at a lesser rate than the fare amount.
5. There are many who have tipped in the order of $0,$5,$10,$15,$20,$25,$30 (increasing in 5 steps)

6. From the square heat map, it is seen that over 50000 the data points lie in the range $5 < fare_amount <$10 ; $2 < tip_amount < $3 (approximate figures)

4. Olive Oil

Data: `olives` dataset in **extracat** package

a. Draw a scatterplot matrix of the eight continuous variables. Which pairs of variables are strongly positively associated and which are strongly negatively associated?

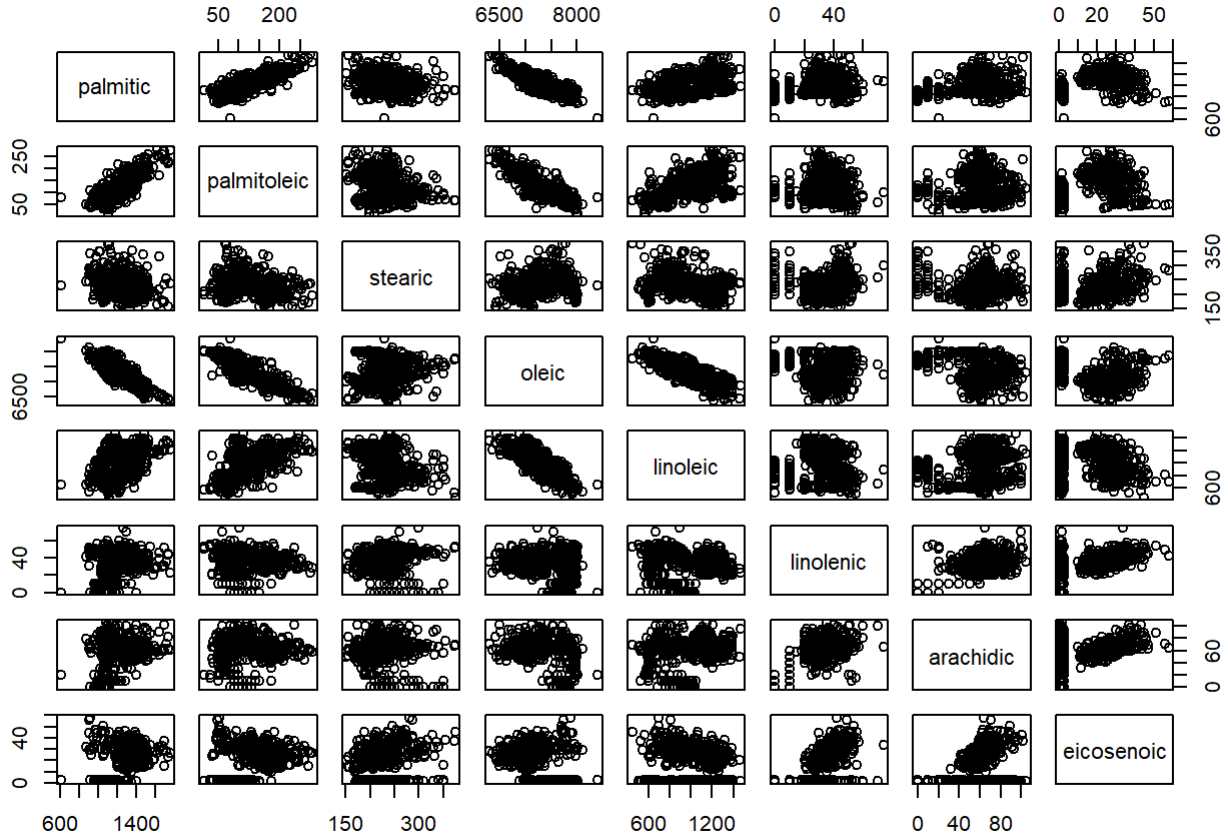
Strongly positively associated: 1) palmitic - palmitoleic

Strongly negatively associated: 1) palmitic - oleic 2) palmitoleic - oleic 3) oleic - linoleic

```
print(levels(olives$Region))
```

```
## [1] "North"    "Sardinia" "South"
```

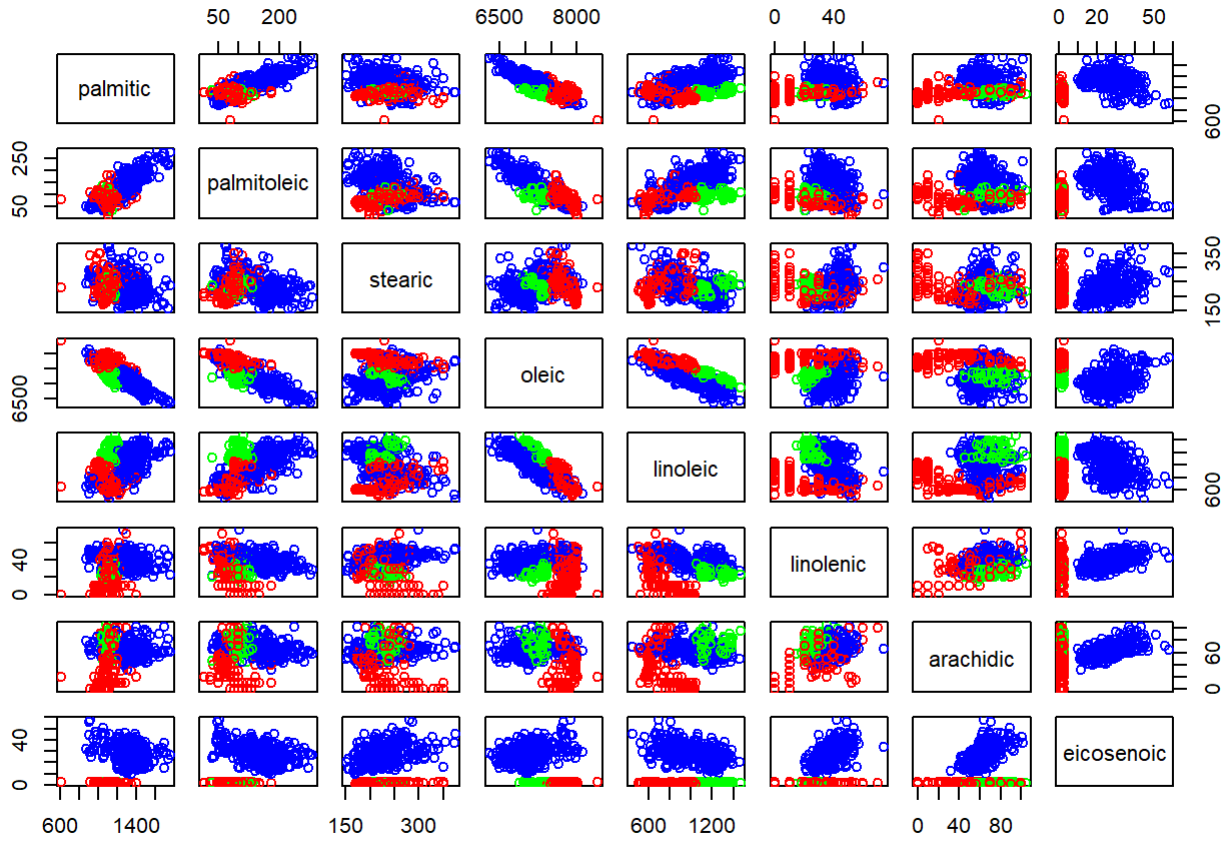
```
toplot <- olives %>% select(palmitic,palmitoleic,stearic,oleic,linoleic,linolenic,arachidic,eicosenoic)
plot(toplot)
```



b. Color the points by region. What do you observe?

On coloring by region, it is seen that maximum contribution to the correlation between any two acids is by South region. If not for the south region, the correlation metrics concluded above will not hold. The data points in other regions are concentrated in single area and aren't showing linear relationship.

```
#North = Red Sardinia = Green South = Blue
color = c("red","green","blue")[olives$Region]
pairs(toplot,col=color)
```



5. Wine

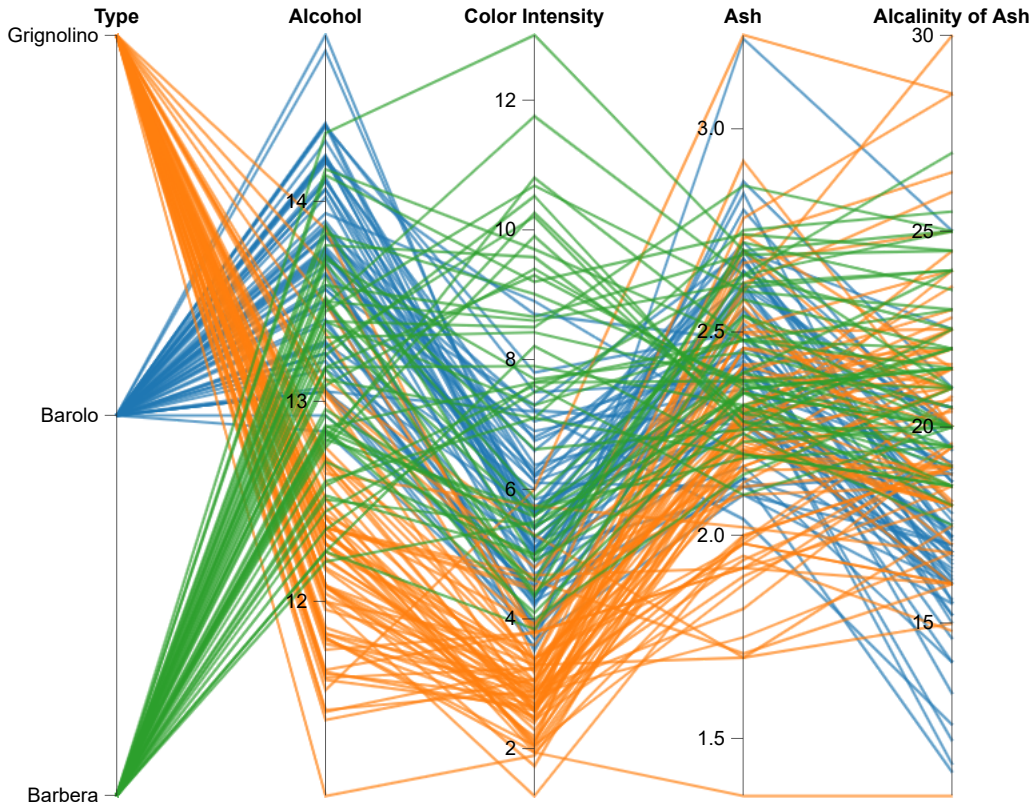
Data: `wine` dataset in **pgmm** package

(Recode the `Type` variable to descriptive names.)

```
data(wine)
my_wine <- wine
my_wine$Type <- recode(my_wine$Type, `1` = "Barolo",`2` = "Grignolino",`3`="Barbera")
```

a. Use parallel coordinate plots to explore how the variables separate the wines by `Type` . Present the version that you find to be most informative. You do not need to include all of the variables.

```
# parcoords function referred from lecture slides
my_wine %>% select(1,2,20,9,10) %>%
parcoords(
    rownames = F
    , brushMode = "1D-axes"
    , reorderable = T
    , queue = T
    , color=list(colorBy="Type",colorScale = htmlwidgets::JS("d3.scale.category10()"))
    )
```



b. Explain what you discovered.

Interactive parallel coordinate plot distinguishes the type of wine by having variable values as given below. I chose three variables to describe the classification.

The alcohol content ranges for various wine types are: Grignolino : 0 to 13.9 Barolo : 12.9 to 15 Barbera : 12.2 to 14.5

The color intensity ranges for various wine types are: Grignolino : 0 to 6 Barolo : 4 to 9 Barbera : 4 to 14

The ash ranges for various wine types are: Grignolino : 1.5 to 3.5 Barolo : 2 to 3.5 Barbera : 2.15 to 2.75

The alkalinity of ash ranges for various wine types are: Grignolino : 0 to 30 Barolo : 0 to 25 Barbera : 17.5 to 27.5

Note: I couldn't find combination of variables that distinctly categorize the wines by type. Hence I explained whatever I understood from the graph above.